5

## What is claimed is:

1. A method in a computer system for improving data transmission of markup language documents, wherein the markup language documents include markup tags and information characters, the method comprising:

converting markup tags in the document to tokens;

creating a token stream;

compressing the token stream using a compression algorithm;

decompressing the token stream using the compression algorithm; and

recreating the markup tags from the token stream.

2. The method as recited in claim 1, wherein converting markup tags to tokens includes:

parsing the document to recognize the tags;

accessing a table of tags, wherein the table lists the tags and their associated tokens;

and

replacing the tags with the tokens.

- 3. The method as recited in claim 2, wherein parsing is performed by a recursivedescent parser.
  - 4. The method as recited in claim 2, wherein the table of tags is extensible.

- 5. The method as recited in claim 1, wherein the markup tags are hypertext markup language tags.
- 6. The method as recited in claim 1, wherein the markup tags are extensible markup language tags.
- 7. The method as recited in claim 1, wherein the token stream includes the converted markup tags.
- 8. The method as recited in claim 1, wherein the token stream includes one or more information characters.
- 9. The method as recited in claim 8, wherein each of the information characters are tokens.
- 10. The method as recited in claim 1, wherein compressing the token stream includes encoding the token stream using Huffman coding.
- 11. The method as recited in claim 1, wherein compressing the token stream includes encoding the token stream using arithmetic coding.

- 12. The method as recited in claim 1, wherein compressing the token stream includes encoding the token stream using LZ77 coding.
- 13. The method as recited in claim 1, wherein compressing the token stream includes encoding the token stream using LZ78 coding.
- 14. The method as recited in claim 1, wherein compressing the token stream includes encoding the token stream using LZW coding.
- 15. The method as recited in claim 1, wherein converting the markup tags, creating a token stream and compressing the token stream are performed on a computer sending the markup language document.
- 16. The method as recited in claim 1, wherein converting the markup tags, creating a token stream and compressing the token stream are performed on a web browser.
- 17. The method as recited in claim 1, wherein converting the markup tags, creating a token stream and compressing the token stream are performed on a transmission network.
- 18. The method as recited in claim 1, wherein decompressing the token stream and recreating the markup tags are performed on a computer receiving the markup language document.

- 70 -

- 19. The method as recited in claim 1, wherein decompressing the token stream and recreating the markup tags are performed on a web browser.
- 20. The method as recited in claim 1, wherein decompressing the token stream and recreating the markup tags are performed on a transmission network.
- 21. A method in a computer system for improving data transmission of markup language documents, the method comprising:

converting markup tags to tokens; and creating a token stream.

22. The method as recited in claim 21, wherein converting markup tags to tokens includes:

parsing the document to recognize the tags;

accessing a table of tags, wherein the table lists the tags and their associated tokens;

and

replacing the tags with the tokens.

23. The method as recited in claim 22, wherein parsing is performed by a recursive-descent parser.

- 24. The method as recited in claim 22, wherein the table of tags is extensible.
- 25. The method as recited in claim 21, wherein the markup tags are hypertext markup language tags.
- 26. The method as recited in claim 21, wherein the markup tags are extensible markup language tags.
- 27. The method as recited in claim 21, wherein the token stream includes the converted markup tags.
- 28. The method as recited in claim 21, wherein the token stream includes one or more information characters.
- 29. The method as recited in claim 28, wherein each of the information characters are tokens.
- 30. The method as recited in claim 21 further comprising compressing the token stream.
- 31. The method as recited in claim 30, wherein compressing the token stream includes encoding the token stream using Huffman coding.

- 72 -

- 32. The method as recited in claim 30, wherein compressing the token stream includes encoding the token stream using arithmetic coding.
- 33. The method as recited in claim 30, wherein compressing the token stream includes encoding the token stream using LZ77 coding.
- 34. The method as recited in claim 30, wherein compressing the token stream includes encoding the token stream using LZ78 coding.
- 35. The method as recited in claim 30, wherein compressing the token stream includes encoding the token stream using LZW coding.
- 36. The method as recited in claim 30, wherein converting the markup tags, creating a token stream and compressing the token stream are performed on a computer sending the markup language document.
- 37. The method as recited in claim 30, wherein converting the markup tags, creating a token stream and compressing the token stream are performed on a web browser.
- 38. The method as recited in claim 30, wherein converting the markup tags, creating a token stream and compressing the token stream are performed on a transmission network.

- 73 -

- 39. The method as recited in claim 30 further comprising decompressing the token stream.
- 40. The method as recited in claim 39, wherein decompressing the token stream is performed on a computer receiving the markup language document.
- 41. The method as recited in claim 39, wherein decompressing the token stream is performed on a web browser.
- 42. The method as recited in claim 39, wherein decompressing the token stream is performed on a transmission network.